

Melihat dan Menganalisa Sebuah Ecommerce Apakah Mengalami Pertumbuhan, Stagnan, atau Bahkan Mengalami Penurunan

Salah satu metrik yang digunakan untuk mengukur performa bisnis eCommerce adalah aktivitas customer yang berinteraksi di dalam platform eCommerce tersebut. Dalam Tugas 2 ini, Saya menganalisis beberapa metrik yang berhubungan dengan aktivitas customer seperti jumlah customer aktif, jumlah customer baru, jumlah customer yang melakukan repeat order dan juga rata-rata transaksi yang dilakukan customer setiap tahun.

Tahapan Pengerjaan Tugas :

- Menampilkan rata-rata jumlah customer aktif bulanan (monthly active user) untuk setiap tahun
- Menampilkan jumlah customer baru pada masing-masing tahun
- Menampilkan jumlah customer yang melakukan pembelian lebih dari satu kali (repeat order) pada masing-masing tahun
- Menampilkan rata-rata jumlah order yang dilakukan customer untuk masing-masing tahun untuk masing-masing customer terlebih dahulu)
- Menggabungkan ketiga metrik yang telah berhasil ditampilkan menjadi satu tampilan table

1. Monthly Active User per Tahun

Query Query History

```
1 select
2     year,
3     round(avg(mau), 0) as average_mau
4 from (
5     select
6         date_part('year', o.order_purchase_timestamp) as year,
7         date_part('month', o.order_purchase_timestamp) as month,
8         count(distinct c.customer_unique_id) as mau
9     from orders o
10    join customers c on o.customer_id = c.customer_id
11   group by year, month
12 ) subq
13 group by year
```

	year double precision 🔒	average_mau numeric 🔒
1	2016	109
2	2017	3695
3	2018	5338

Langkah pertama adalah membuat suatu tabel sementara dengan *subquery* di *from clause*. Tabel sementara digunakan untuk menghitung jumlah customer unik yang aktif melakukan order setiap bulannya, yang disebut dengan *Monthly Active User* (MAU). Fungsi *date_part* digunakan untuk melakukan ekstrak komponen *timestamp*, dalam kasus ini, komponen *year* (tahun) dan *month* (bulan) yang diekstrak sesuai kebutuhan. Setelah mendapatkan angka MAU setiap bulan, dapat dilakukan agregasi yang lebih besar untuk mendapatkan rata-rata MAU untuk masing-masing tahun, karena itu dilakukan perhitungan *avg* (rata-rata) yang dikelompokkan berdasarkan *year* (tahun).

2. Jumlah Customer Baru per Tahun

Query Query History

```
1 select
2     date_part('year', first_purchase_time) as year,
3     count(1) as new_customers
4 from (
5     select
6         c.customer_unique_id,
7         min(o.order_purchase_timestamp) as first_purchase_time
8     from orders o
9    join customers c on c.customer_id = o.customer_id
10   group by customer_unique_id
11 ) subq
12 group by year
13
```

	year double precision 🔒	new_customers bigint 🔒
1	2016	326
2	2017	43708
3	2018	52062

Pada tabel sementara tersebut berisi informasi tanggal pertama kali order (first purchase timestamp) untuk masing-masing customer. Hal ini dapat dengan mudah didapatkan dengan menggunakan fungsi agregasi MIN pada kolom order_purchase_timestamp untuk mendapatkan tanggal terkecil dari masing-masing customer. Setelah mendapatkan tanggal order pertama untuk setiap customer, maka langkah berikutnya bisa dilakukan ekstrak tahun kemudian menghitung ada berapa jumlah customer untuk masing-masing tahun tersebut. Angka terakhir ini menunjukkan ada berapa banyak jumlah customer baru pada masing-masing tahun yang ada.

3. Customer Repeat Order

Query	Query History
<pre> 1 select 2 year, 3 count(distinct customer_unique_id) as repeating_customers 4 from (5 select 6 date_part('year', o.order_purchase_timestamp) as year, 7 c.customer_unique_id, 8 count(1) as purchase_frequency 9 from orders o 10 join customers c on c.customer_id = o.customer_id 11 group by year, customer_unique_id 12 having count(1) > 1 13) subq 14 group by year 15 </pre>	

	year double precision 🔒	repeating_customers bigint 🔒
1	2016	3
2	2017	1256
3	2018	1167

Pada subtask 3 ini, tabel sementara yang dibuat menunjukkan banyak order yang dilakukan oleh masing-masing customer per tahun. mengelompokkan perhitungan COUNT berdasarkan tahun dan customer_unique_id. tujuannya adalah untuk mendapatkan customer yang melakukan repeat order, menggunakan filter HAVING untuk mendapatkan customer yang melakukan order lebih dari 1. Setelah itu, agregasi COUNT yang dikelompokkan berdasarkan tahun bisa dilakukan untuk menghitung ada berapa banyak customer yang melakukan repeat order (order lebih dari 1) untuk masing-masing tahun.

4. Customers Average Orders

Query Query History

```
1 select
2     year,
3     round(avg(frequency_purchase),3) as avg_orders_per_customers
4 from (
5     select
6         date_part('year', o.order_purchase_timestamp) as year,
7         c.customer_unique_id,
8         count(1) as frequency_purchase
9     from orders o
10    join customers c on c.customer_id = o.customer_id
11   group by year, customer_unique_id
12 ) a
13 group by year
14
```

	year double precision 🔒	avg_orders_per_customers numeric 🔒
1	2016	1.009
2	2017	1.032
3	2018	1.024

Setelah tabel yang berisi informasi frekuensi order untuk masing-masing customer setiap tahun, maka agregasi lebih lanjut bisa dilakukan dengan melakukan AVG pada kolom frekuensi yang kemudian dikelompokkan berdasarkan tahun. Dengan begitu, angka rata-rata frekuensi order untuk masing-masing customer di setiap tahunnya berhasil didapatkan.

5. Hasil Metrik

	year double precision 🔒	average_mau numeric 🔒	new_customers bigint 🔒	repeating_customers bigint 🔒	avg_orders_per_customers numeric 🔒
1	2016	108.67	326	3	1.009
2	2017	3694.83	43708	1256	1.032
3	2018	5338.20	52062	1167	1.024